1. Change directory =🡺 **$cd foldername** (press TAB to complete the name of folder or file)
2. Go back one folder 🡺 **$cd** ..
3. Go to root directory where linux is installed 🡺**$ cd /**
4. Go to main root (this a user inside the linux) 🡺 **$cd**
5. To see all the containing files inside the file 🡺 **$ls**
6. To exit a program 🡺**$exit** or exit()
7. To quit a open file 🡺 **$:q**
8. To quit and save a file 🡺 **$:wq**
9. If you don’t have access to open and close a file 🡺 **$:q!** or **$:wq!**
10. To get information about cpu 🡺 **$goto** **/** THEN **$cd proc** THEN OPEN **$cpuinfo**
11. User is present in🡺 **$/home/username/**
12. root@localhost 🡺 **$username@hostname**
13. to show all the hidden files or folder 🡺 **$ls -a or $ls -al**
14. to see all the information of a file and folder 🡺 **$ls -l**
15. to show files in latest to older view 🡺**$ls -t**
16. combine the letters to make your own commands 🡺**$ ls -lt** (to show info and latest view)
17. display the files in reverse order 🡺 **$ls -r**
18. to check where are we right now (present working directory) PWD 🡺**$pwd**
19. create a new (multiple) directory 🡺**$mkdir foldername1 foldername2 foldername3**
20. if we want to create a directory in another directory from outside 🡺 **$mkdir (FULL\_PATH) /root/desktop/swapnil** 🡺 it will create a swapnil directory inside the root/desktop
21. create a file(multiple) using TOUCH command 🡺 **$touch filename1 filename2**
22. command to remove(DELETE) empty directory 🡺 **$rmdir directoryname1 directoryname2**
23. command to remove(DELETE) entire directory 🡺 **$rm -r directoryname1 directoryname2**
24. command to remove(DELETE) a file from a directory 🡺 **$rm filename**
25. clear the entire command screen 🡺 **$clear**
26. to show what process is happening after any command execute 🡺 -v
27. to give a permission to perform any command 🡺 sudo or -f (example: rm -r -f -v directoryname)
28. copy a file(multiple) from source to destination 🡺 cp source destination (example: cp file.txt file1.txt file2.txt foldername) [if you are in a directory and you want to copy a file from that directory then you don’t have to give entire path..just give filename]
29. copy a file(multiple) in the directory you are right now 🡺 cp source . (example: cp file1.txt file2.txt .)
30. to copy a directory in to other directory 🡺 cp -r source destination (example cp -r dir1 dir2)
31. to move a file(multiple) to any directory🡺mv source destination
32. to move a directory to other directory 🡺 mv source destination
33. to give a path to a directory which is in earlier directory 🡺 ../../
34. rename a file name or directory 🡺 mv filename newfilename 🡺 mv foldername newfoldername
35. to read the contents of file (it will show everything the file contains) 🡺 cat filepath(FILENAME)
36. creating a file along with data in the file (write something into the file along with creating a file)🡺 cat > filename.extention [example: cat > swap.html ENTER WRITE WHATEVER THAT YOU WANT TO ADD IN THE FILE] [AFTER THIS PRESS CONTROL + C TO SAVE]
37. to copy the content of a file to another new file🡺 cat source\_file\_path destination\_filename(with extention) [example: cat /home/swapnil/log swapnil.txt] [this will create a file swapnil.txt and content from log file will copy to this swapnil.txt file]
38. to go to the file to view content inside the file 🡺 less filename(path) [using this we will go to inside the file][if the file has more than one page then using control + f we can go to next page]
39. pipe => this is used to get the output of previous command and make input to the next command🡺 ls | cat [this will get all the data that ls has and consider it as input for cat command then this cat command shows the input in shell][example: ls | less] [this will open a file and it will display all the file that ls is showing]
40. > (redirect command)=> this command is used to insert whatever before < to the next command (filename) [example: hello world > flaskapp.html (this will create a html file that contain hello world in it)][example: ls > new.txt ( this new file contain whatever ls give (output))]
41. To make a zip file ie to archieve all the file in single folder or zip file 🡺 tar -cvf filename.tar directory\_that\_need\_to\_archieve [example: tar -cvf backup.tar /home/swapnil/flaskapp/]( this will create a tar file that contain all the file from flaskapp to backup.tar file)
42. How to get all the archived file again🡺tar -xvf backup.tar [this will give back all the file into the directory you are right now]
43. How to see the contents of archived file 🡺tar -tvf backup.tar [this will show the content of backup.tar]
44. How to add a file into the tar file 🡺 tar -rvf backup.tar filename.txt [this will insert a filename.txt in the backup.tar]
45. Create a soft link (this will direct to the directory just like windows link[this will not contain any data]) 🡺 ln -s file\_path\_that­\_is\_linked file\_name\_for\_link [example: ln -s /root/swapnil/swap.txt /root/flaskapp/swap-link [this will create a shortcut swap-link in flaskapp folder]
46. To unlink the softlink 🡺 unlink file\_path [you got be in that folder or need to give path to the file] [example: unlink /root/swap-link ]
47. To create a hard link 🡺 ln file\_path\_for\_source filepath\_destiantion\_filename [example: ln /root/swapnil/swap.txt /root/desktop/swap-link] [hard link has the same properties as the original file if we make any changes here the changes will make in original file]

EXTRA COMANDS FOR KNOWLEDGE  
$ add . or $scp . 🡪 this **.** is use for selecting all file inside the folder